



Halle, 03.. Juli 2006

Programmiersprachen (SS 2006)

Übungsserie 13

Aufgabe 1 (Mechanismen in Java)

Begründen Sie kurz Ihre Antworten auf folgende Fragen:

- Kann Überschreiben in derselben Klasse stattfinden?
- Kann Überladen in derselben Klasse stattfinden?
- Kann Überschreiben in einer abgeleiteten Klasse stattfinden?
- Kann Überladen in einer abgeleiteten Klasse stattfinden?

Aufgabe 2 (Liskovsches Ersetzungsprinzip)

- Wie lautet das Liskovsche Ersetzungsprinzip für das Erweitern?
Verwenden Sie die Begriffe Objekt, Vater-Objekt für Ihre Antwort.
- Wie lautet das Liskovsche Ersetzungsprinzip für das Überschreiben?

Aufgabe 3 (Mehrfachvererbung, C++, Eiffel)

C++ bietet Mehrfachvererbung unter Anwendung einer Liste von Basisklassen, die durch Kommata voneinander getrennt sind, wie in

```
class C : public A, private B { ... };
```

Mehrfachvererbung in C++ erzeugt gewöhnlich getrennte Kopien für jede Klasse auf einem Vererbungspfad, die sogenannte *wiederholte* Vererbung.

Geben Sie den Vererbungsgraphen für folgende Deklaration an:

```
class A { .. };  
class B : public A { .. };  
class C : public A { .. };  
class D : public B { .. }, public C { ... };
```

Um nur eine Kopie von A in der Klasse D zu erhalten, muß die Vererbung unter Verwendung des Schlüsselworts `virtual` deklariert werden:

```
class A { .. };
class B : virtual public A { .. };
class C : virtual public A { .. };
class D : public B { .. }, public C { ... };
```

Geben Sie für diese *gemeinsame* Vererbung den Vererbungsgraphen an.

In Eiffel ist standardmäßig die *gemeinsame* Vererbung eingestellt.

```
class A ... ;
class B ... inherit A ...
class C ... inherit A ...
class D ... inherit B ... ; inherit C ...
```

Wiederholte Vererbung, bei der mehrere Kopien eines Vorfahrens in einem Nachkommen enthalten sind, kann durch Umbenennung erhalten werden:

```
class A ... ;
class B ... inherit A ...
class C ... inherit A ...
class D ... inherit B ... rename...;
- - gib jedem Feature f von A den Namen B_f
inherit C ... rename ...;
- - gib jedem Feature f von A den Namen C_f
```

Angenommen, Klasse D erbt sowohl von der Klasse B als auch von der Klasse C, und die Klassen B und C erben beide eine Methode `m` von Klasse A, aber B definiert `m` neu. Wenn `d` ein Objekt der Klasse D ist, dann kann `d.m` entweder `B.m` oder `A.m` sein.

- Beschreiben Sie, wie Eiffel dieses Problem löst.
- Gibt es eine äquivalente Lösung in C++?

Aufgabe 4 (Vererbung und dynamisches Binden)

In Eiffel kann ein Programmierer nicht deklarieren, ob eine Methode während der Ausführung dynamisch zu binden ist oder nicht (wie *virtual* in C++).

- Bedeutet dies, daß alle Methoden in der Laufzeitumgebung als *virtual* angenommen werden müssen? Warum?
- Ist es möglich, in Eiffel statische Bindung zu erhalten?

Besprechung ist am 04. Juli 2006.